HADMÉRNÖK
VÉDELEMINFORMATIKA

17. évfolyam (2022) 4. szám 153–174.
DOI: 10.32567/hm.2022.4.10

István Paráda,[1] András Tóth[2]

Possible Scenario for Malware Exploit Investigation with Data-Driven Architecture[3]

In this article, the authors present a data-driven architecture-based malware exploit analysis as the next part of the Penetration Testing article series. The analysis contributes greatly to investigating malicious attacks, which are becoming increasingly sophisticated in cyberspace, thus posing a significant threat to the information and communication networks of state and non-state actors. To achieve their research objectives, the authors use analytical evaluation methods to define the principles, modular elements and procedures of the data-driven architecture to be applied, where decisions are made based on the available data. On this basis, they have presented an analytical process that can help the public and defence sectors to analyse this type of attack, thus facilitating recovery processes.

Keywords: **Metasploit, Metasploit Framework, vsFTPd, NMAP, TCP, FTP**

Thanks to increasingly sophisticated protection, logging and analysis techniques, we have much more information available to investigate an incident. There are a few basic reasons why there has recently been so much emphasis on data-driven information. First, technological advances in computing and networking capacity have made it possible to publish and transmit unprecedented amounts of data. Second, technological advances in artificial intelligence have helped us analyse these vast amounts of data in ways that were impossible before. These two factors lead to many cases where a machine can draw conclusions from the data and make (better) decisions based on the results.

In response to the volume and sophistication of malware, security experts rely on data-driven architecture analysis to detect malicious activities and software. Data-driven architectural analysis is the process of running binary patterns by experts to produce reports that summarise their real runtime behaviour. These reports can be

[1] PhD student, University of Public Service, e-mail: paradaistvan@gmail.com

[2] Associate Professor, University of Public Service, e-mail: toth.hir.andras@uni-nke.hu

[3] The research was financed by the National Research Development and Innovation Office Fund and was implemented in the funding of the Thematic Program of Excellence 2020 application program number TKP2020-NKA-09.

used to identify malware and determine attributes of threat types. They are crucially important in the government and defence sectors, where there are many critical information infrastructures, the loss of which could seriously compromise the functioning of the state or one of its critical infrastructures. Therefore, frameworks and procedures to prevent and deter malicious activities in these areas are of paramount importance.

In preparing this paper, the authors' main goal was to develop a framework to help investigate the increasingly sophisticated malicious attacks in cyberspace. For this purpose, the authors used analytical evaluation methods to define the principles, modular elements and procedures of the data-driven architecture to be applied. Data-driven means that decisions are made based on the available data.

This study develops a framework for malware detection and threat family identification using supervised machine learning techniques. The developed framework can support the work of professionals performing tasks in e-government, state and non-state actors' digitisation in cybersecurity incident detection and recovery. The results show the efficiency and portability of our solutions across a wide range of analyses and settings.

To achieve the best results, the authors chose Elasticsearch software, a real-time technology that allows working with all volumes with different APIs (from gigabytes to petabytes). Besides Kibana, many different solutions can take advantage of the open APIs offered by Elasticsearch and build visualisations on the resulting data, but Kibana is the only technology dedicated to this.

1. Data-driven architecture

Data can typically be anything. Accordingly, we can talk about business data, infrastructure data, accounting data, structured or unstructured data and personal data. For any organisation (public or non-public), extracting the value of data from huge data sets is a huge challenge, which helps to extract useful information from the data. This is typically challenging due to the following factors:

- Data complexity: Huge amounts of data, mostly from many different data sources and containing much useless information (noise).
- Data from various sources: Data can come from many sources that are not relevant to an organisation. For example, they can be legacy systems or databases, infrastructures, tools, or applications that are irrelevant to the organisation. These should be continuously monitored and validated.
- The volume of data is growing very fast: Managing it is a major challenge for all organisations. Therefore, particular attention must be paid to scaling the data management infrastructure to make it easier to determine which data should be retained.

In their paper, Wang et al. discussed data-driven architectures in 5G as the communication part of critical infrastructures. In the article, they formulated the following requirements that the architecture should meet:

- The architecture must monitor the applications used by users and the Quality of Service (QoS) status in real-time.
- The architecture must maintain a data mining system that can predict user preferences/expectations for the applications used.
- The architecture should manage communication resources based on the QoS state and predicted preferences/expectations to maintain a satisfactory Quality of Experience (QoE).[4]

Numerous features can be added to this in the design of public and defence communication systems. There, special attention must be paid to the management of sensitive data, so it is not enough to monitor and analyse user applications but also to pay special attention to solutions such as endpoint protection, encryption procedures, intrusion prevention and detection services. In principle, they are not only present in communication networks but must be applied to all systems using information and communication technologies (ICT). Accordingly, they should be applied to all areas of the public and defence sectors, such as smart cities,[5] power grid systems,[6] industrial control systems[7] and healthcare, among others.

The elements of the data-driven architecture that support the above requirements to be fully met are important. In the architecture case, data transport, data ingest, data storage at scale and data visualisation play a key role.

1.1. Data ingest

The data ingestion layer is responsible for receiving data, which includes commonly used transport protocols and data formats, while providing the ability to extract and transform data before final storage. From our perspective, data processing is the extraction, transformation, and loading of data, often referred to as the input pipeline, and essentially receives data from the transport layer to push it into a storage layer. It has these functions:

- In general, the ingestion layer has a pluggable architecture to facilitate integration with different data sources and destinations, using a set of plugins. Some of the plug-ins are designed to receive data from senders, which means that the data does not always come from the sender and can be delivered directly from a data source such as a file, a network or even a database.
- The data ingestion layer is used to prepare data, for example by analysing, formatting, correlating data with other data sources, and normalising and enriching data before storage. There are many improvements, but the most important is that it improves the quality of the data, providing better observations for visualisation.

[4] Wang et al. 2017
[5] Fang et al. 2021
[6] Jia et al. 2018
[7] Wang et al. 2018

- Data input and transformation consume computational resources. It is essential to take this into account, usually in terms of maximum data throughput per unit, and to plan the load by distributing the input over several data input instances. This is an essential aspect of real-time, or more precisely near real-time, visualisation.

1.2. Data shipping

The architecture must be able to transport any structured or unstructured data/event; in other words, it transports data from remote machines to a central location. This is usually done by a lightweight agent deployed on the same machine as the data sources or, in different aspects, on a remote machine:

- Lightweight because, ideally, it should not compete for resources with the actual process that produces the data; otherwise, it may reduce the expected performance of the process.
- There are many data transport technologies; some are tightly coupled to a specific technology; others are based on an extensible framework that is relatively adaptable to the data source.
- Data transport is not only about sending data over the wire but also about security and ensuring that the information is delivered to the right destination via an end-to-end secured pipeline.
- Another aspect of data transport is the management of data loading. Data transport must be done in proportion to the load the end destination can accommodate; this function is called backpressure management.

1.3. Storing data at scale

This ensures the basic, long-term preservation of data. In addition, it provides the essential functionality to search, analyse and discover insights into the data.

The storage layer generally provides:

- Scalability is the main aspect, with storage used for different data volumes, starting from gigabyte (GB), terabyte (TB) and petabyte (PB).
- A non-relational and highly distributed data store is usually used, allowing fast data access and analysis on large volumes and different data types, namely a NoSQL data store.
- For data visualisation, the repository must publish an application programming interface (API) for data analysis. Allowing the visualisation layer to perform statistical analysis, such as grouping data by a given dimension, would not scale.

1.4. Visualising data

In a data-driven architecture, the visualisation layer is one of the potential data consumers and mostly focuses on bringing key performance indicators (KPIs) to the stored data. It has the following basic functions:

- It should be lightweight and only display the result of the processing done in the storage layer.
- Allow the user to explore the data and quickly get out of the box.
- It brings a visual way to ask unexpected data questions, rather than having to perform a corresponding prompt.
- Modern data architectures need to meet accessibility needs.
- KPIs should be as fast as possible, and the visualisation layer should display data in near real-time.
- The visualisation framework should be extensible and allow users to customise existing tools or add new functionality depending on their needs.

2. Overview of the elastic stack

The Elastic stack, formerly called ELK from the acronym of three open-source projects: Elasticsearch, Logstash and Kibana, ensures the different layers needed to implement a data-driven architecture. The first is the ingestion layer with Beats and Logstash, the second is a distributed data store with Elasticsearch and the third is the visualisation with Kibana.

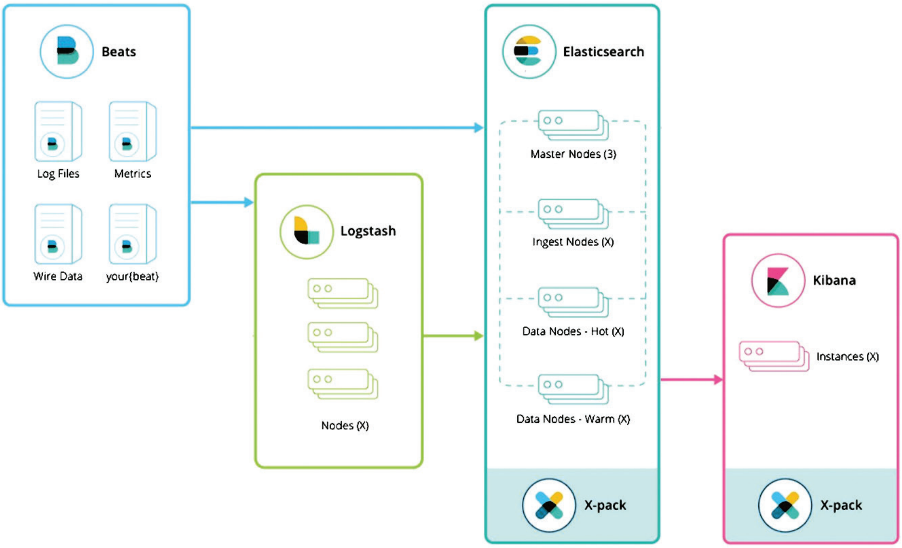


Figure 1: The elastic stack structure
Source: Azarmi 2017

2.1. Elasticsearch

Elasticsearch is a distributed and scalable data store from which Kibana pulls all aggregation results used in the visualisation. It is flexible and scalable by nature, so nodes can be added to the Elasticsearch cluster very easily, depending on the needs. Furthermore, Elasticsearch is a highly available technology, which means that:

First, data is replicated within the cluster, so at least one copy of the data is preserved in the event of a failure.

Secondly, due to its distributed nature, Elasticsearch can distribute the indexing and search load across the cluster nodes, ensuring service continuity and service level agreement (SLA) compliance.

Structured and unstructured data can be handled, and as the data is visualised in Kibana, it is noticeable that the data, or using Elasticsearch's vocabulary, documents are indexed in JavaScript Object Notation (JSON) documents. In addition, JSON makes it very practical to handle complex data structures as it supports nested documents, arrays, etc.

Elasticsearch is a developer-friendly solution that offers several REST APIs for interacting with data or cluster settings. Documentation for these APIs can be found at www.elastic.co/guide/en/elasticsearch/reference/current/docs.html.

In addition to these APIs, client APIs allow Elasticsearch to integrate with most technologies such as Java and Python.

Kibana generates the requests to the cluster for each visualisation. The final key aspect of Elasticsearch is that it is a real-time technology that allows working on volumes ranging from gigabytes to petabytes using a variety of APIs. In addition to Kibana, several other solutions can leverage the open APIs offered by Elasticsearch to build visualisations on top of data; but Kibana is the only technology dedicated to this.[8]

2.2. Beats

Beats is a lightweight data transporter that delivers data from various sources, such as applications, end devices, or networks. Beats is built on an open-source library that allows the beat to send data to Elasticsearch, as shown in the following image.

The diagram shows the following Beats:

- Packetbeat essentially looks for packets over the network wire for certain protocols such as MySQL and HTTP. It captures all the basic metrics used to monitor a given protocol. For example, HTTP receives the request and response, wraps them in a document, and indexes them in Elasticsearch.
- Filebeat is used to safely transport the contents of a file from point A to point B in a similar way to the tail command. This beat can be used with the new

[8] GoLinuxCloud 2020

ingest node (www.elastic.co/guide/en/elasticsearch/reference/master/ingest.html) to transfer data directly from the file to Elasticsearch, which processes it before indexing.

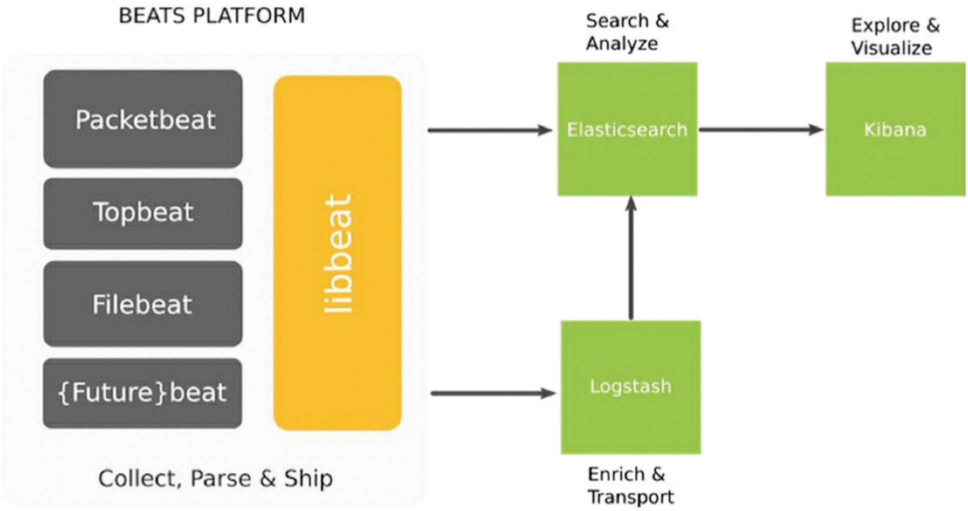


Figure 2: The Beats architecture
Source: Azarmi 2017

2.3. Ingestion pipeline without ingest

As shown in the previous figure, the data is first delivered by Beats, then sent to a message broker, after which it is processed by Logstash and indexed by Elasticsearch. The disadvantage of Beats is that it has some basic filtering functions, but these do not provide the level of transformation that Logstash can provide.

2.4. Ingestion pipeline with Ingest node

As shown in Figure 2, the architecture is reduced to two components using the filebeat and ingest node, and then the content is rendered in Kibana. To send machines or bapplication execution samples to Elasticsearch, we can use Topbeat, the first Metricbeat that allows us to do this. We also used this solution to transport our applied computer data and visualise it in Kibana during the test. A huge advantage of this solution is that this beat comes with pre-made templates that are standardised; accordingly, the templates received just need to be imported into Kibana for visualisation.

While Beats does offer some basic filtering features, they do not offer the level of conversion that Logstash does.

2.5. Logstash

Logstash is a data processor that uses a centralised data processing paradigm. It allows users to collect, enrich/transform and deliver data to destinations using more than 200 extensions, as shown in Figure 3.

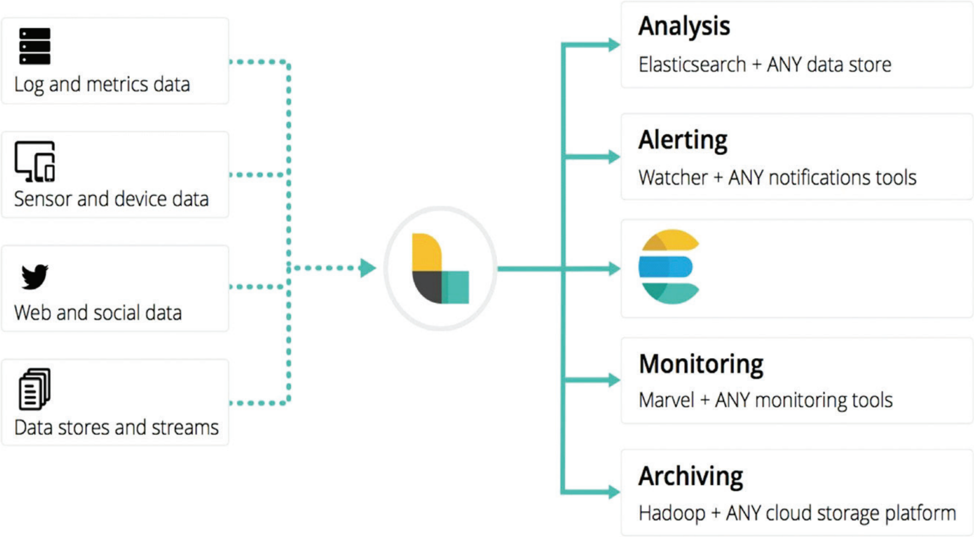

Log and metrics data
Sensor and device data
Web and social data
Data stores and streams
Analysis
Elasticsearch + ANY data store
Alerting
Watcher + ANY notifications tools
Monitoring
Marvel + ANY monitoring tools
Archiving
Hadoop + ANY cloud storage platform

Figure 3: Logstash, the processing pipeline
Source: Azarmi 2017.

Logstash can collect data from several sources, one of which is Beats, as the out-of-the-box integration of Logstash is included in every Beat. However, in this case, the roles are separated clearly: Beats is responsible for delivering the data by default, while Logstash enables data processing before indexing. Consequently, Logstash should be used to prepare the data during the visualisation process.

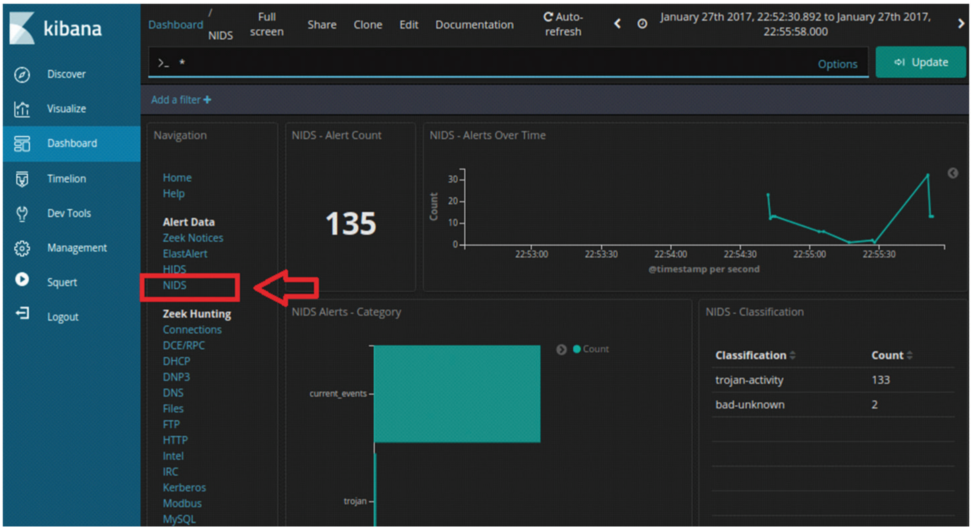
2.6. Kibana

Kibana is where all the operations of the user interface take place. Most visualisation technologies handle the analytical processing, while Kibana is just a web application that displays the analytical processing done by Elasticsearch. It does not load data from Elasticsearch and then process it but leverages the power of Elasticsearch to do all the heavy lifting. This enables real-time visualisation at scale: as the data grows, the Elasticsearch cluster scales relatively to offer the best latency as a function of SLAs. In addition, Kibana provides visual performance for Elasticsearch aggregations, allowing time-series datasets or segmentation of data fields to be sliced as easily as possible. Kibana is equipped with time-based visualisation, even if the data can arrive without a timestamp, and brings visualisation built for Elasticsearch aggregation framework visualisation.

3. Explore malware exploit by using Kibana

In investigations, it is important to determine when the attack occurred. Preliminary information on this can be provided by, for example, the Snort Network Intrusion Detection and Prevention System (NIDS), which can detect intrusion-based attacks. Snort is the world's foremost Open Source Intrusion Prevention System (IPS). Snort IPS uses several different rules in its analyses to help determine malicious network activity, and in its investigations, it can identify the unexpected packets that may be causing this malicious activity and send alerts to users immediately.

In the case of a malware exploit scan, it is very important to have an accurate timeframe, which can be achieved by narrowing down the timeframe. Therefore, the first step should definitely be to set an absolute time interval in Kibana to narrow the focus to the log data that is important to you. In this case, we get a graph showing a single entry. To see more detail, we need to restrict the time interval further to be examined and displayed. Once the time range has been reduced to what we need, we can then sort the events by their occurrence, and then analyse the details of each event by the time they occurred. Figure 4 shows how Kibana displays the total number of NIDS Alerts in the dashboard interface.



Classification	Count
trojan-activity	133
bad-unknown	2

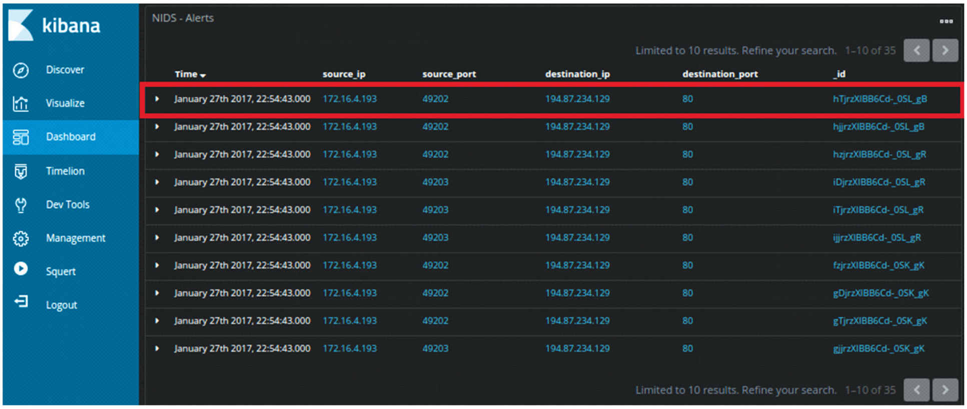
Figure 4: Total number of NIDS alerts in Kibana

Source: Compiled by the authors based on application Kibana.

Examining the details of the extended warning, we can get, amongst others, the following results:

- the date of the first detected NIDS alert in Kibana
- the IP address of the source in the alert
- destination IP address in the alert
- the destination port and service in the alert
- the classification of the alert
- the geographical name of the destination country

The visualisation of the above results in Kibana is illustrated in Figure 5, where the time of the first alert, the source and destination IP address and port are shown.



Limited to 10 results. Refine your search. 1–10 of 35

Time	source_ip	source_port	destination_ip	destination_port	_id
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	hTjrzXIBB6Cd-_0SL_gB
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	hjjrzXIBB6Cd-_0SL_gB
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	hzjrzXIBB6Cd-_0SL_gR
January 27th 2017, 22:54:43.000	172.16.4.193	49203	194.87.234.129	80	iDjrzXIBB6Cd-_0SL_gR
January 27th 2017, 22:54:43.000	172.16.4.193	49203	194.87.234.129	80	iTjrzXIBB6Cd-_0SL_gR
January 27th 2017, 22:54:43.000	172.16.4.193	49203	194.87.234.129	80	ijjrzXIBB6Cd-_0SL_gR
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	fzjrzXIBB6Cd-_0SK_gK
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	gDjrzXIBB6Cd-_0SK_gK
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	gTjrzXIBB6Cd-_0SK_gK
January 27th 2017, 22:54:43.000	172.16.4.193	49203	194.87.234.129	80	gjjrzXIBB6Cd-_0SK_gK

Limited to 10 results. Refine your search. 1–10 of 35

Figure 5: The results of NIDS alerts in Kibana
Source: Compiled by the authors based on application Kibana.

In the web browser of an internet-connected computer, we went to the link in the signature_info field of the alert. This led us to the Emerging Threats Snort alert rule for the exploit. A set of rules is shown. This is because signatures may change over time, or new and more specific rules may be developed. The most recent rule is at the top of the page.

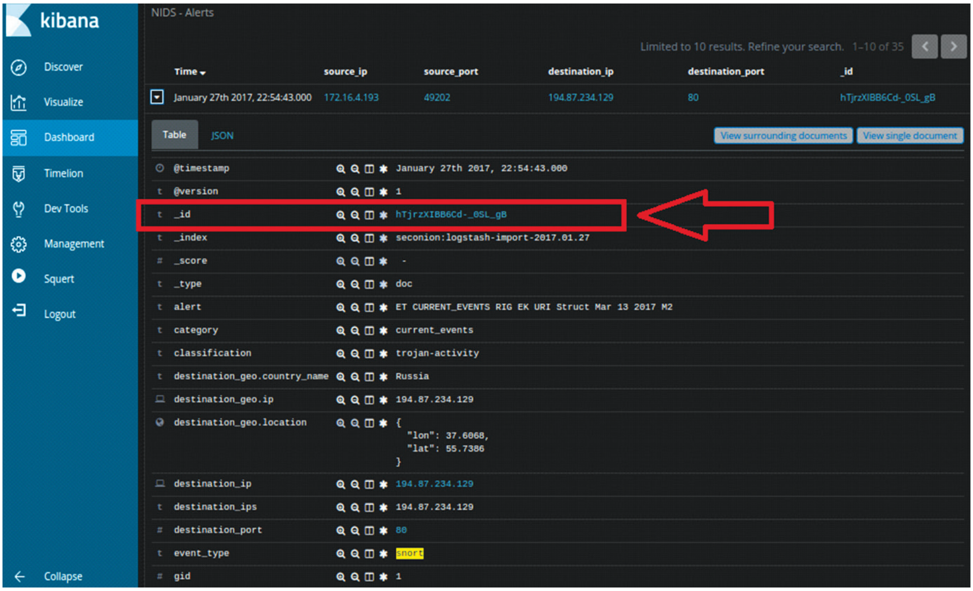
Examining the details of the rule, we came to the following conclusions:

- the malware family for this event is Exploit_Kit_RIG
- the severity of the exploit is the signature severity, which is Major

We then defined what an Exploit Kit is. An Exploit Kit is a malware that aims to infect user devices or network elements with malicious software that uses multiple websites and redirects to achieve its goal. Exploit Kits often use a so-called drive-by method to initiate the attack process. In this type of attack, the user opens a site that appears to be secure, but which contains vulnerabilities that the attackers are aware of and can easily exploit. The vulnerabilities make it much easier for the threat actors to operate, as they allow them to insert their malicious code into the HTML code of the website. The code is often inserted into an iFrame, which allows content from different web pages to be displayed on the same web page. In most cases, attackers create an invisible iFrame that links the browser to a malicious website. In addition, the HTML loaded into the browser from the website often contains JavaScript that sends the browser to another malicious website or downloads malware to the computer.[9]

3.1. Transcript CapME!

Clicking on the _id of the alert will switch to CapME! to examine the transcript of the event, which is shown in Figure 6.



Time	source_ip	source_port	destination_ip	destination_port	_id
January 27th 2017, 22:54:43.000	172.16.4.193	49202	194.87.234.129	80	hTjrzXIBB6Cd-_0SL_gB

@timestamp	January 27th 2017, 22:54:43.000
@version	1
_id	hTjrzXIBB6Cd-_0SL_gB
_index	seconion:logstash-import-2017.01.27
_score	-
_type	doc
alert	ET CURRENT_EVENTS RIG EK URI Struct Mar 13 2017 M2
category	current_events
classification	trojan-activity
destination_geo.country_name	Russia
destination_geo.ip	194.87.234.129

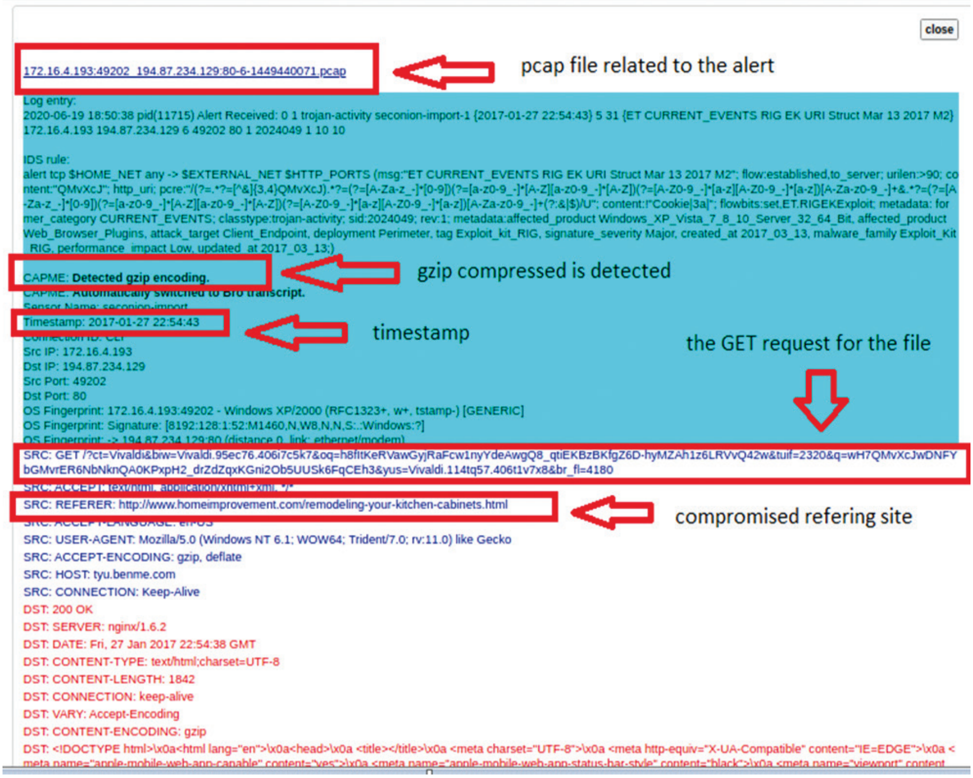
destination_geo.location	{
	"lon": 37.6068,
	"lat": 55.7386
	}

destination_ip	194.87.234.129
destination_ips	194.87.234.129
destination_port	80
event_type	snort
gid	1

Figure 6: The CapME! window in Kibana
Source: Compiled by the authors based on application Kibana.

Further analysing the results shown in the CapME! window, the session transcript shows that the transactions between the source computer and the destinations reached by the source computer are highlighted in blue. The transcript also contains several valuable information, including a link to the pcap file associated with the alert. These results shown in Figure 7.

[9] O'Driscoll 2019



close
pcap file related to the alert
gzip compressed is detected
timestamp
the GET request for the file
compromised refering site

```
172.16.4.193:49202_194.87.234.129:80-6-1449440071.pcap
Log entry:
2020-06-19 18:50:38 pid(11715) Alert Received: 0 1 trojan-activity seconion-import-1 {2017-01-27 22:54:43} 5 31 {ET CURRENT_EVENTS RIG EK URI Struct Mar 13 2017 M2}
172.16.4.193 194.87.234.129 6 49202 80 1 2024049 1 10 10

IDS rule:
alert tcp $HOME_NET any -> $EXTERNAL_NET $HTTP_PORTS (msg:"ET CURRENT_EVENTS RIG EK URI Struct Mar 13 2017 M2"; flow:established,to_server; urilen:>90; co
ntent:"QMvXcJ"; http_uri; pcre:"/(?=.*?=[^&]{3,4}QMvXcJ).*?=(?=[A-Za-z_-]*[0-9])(?=[a-z0-9_-]*[A-Z][a-z0-9_-]*[A-Z])(?=[A-Z0-9_-]*[a-z][A-Z0-9_-]*[a-z])[A-Za-z0-9_-]+&.*?=(?=[A
-Za-z_-]*[0-9])(?=[a-z0-9_-]*[A-Z][a-z0-9_-]*[A-Z])(?=[A-Z0-9_-]*[a-z][A-Z0-9_-]*[a-z])[A-Za-z0-9_-]+(?:&|$)/U"; content:!"Cookie|3a|"; flowbits:set,ET.RIGEKExploit; metadata: for
mer_category CURRENT_EVENTS; classtype:trojan-activity; sid:2024049; rev:1; metadata:affected_product Windows_XP_Vista_7_8_10_Server_32_64_Bit, affected_product
Web_Browser_Plugins, attack_target Client_Endpoint, deployment Perimeter, tag Exploit_kit_RIG, signature_severity Major, created_at 2017_03_13, malware_family Exploit_Kit
_RIG, performance_impact Low, updated_at 2017_03_13;)

CAPME: Detected gzip encoding.
CAPME: Automatically switched to Bro transcript.
Sensor Name: seconion-import
Timestamp: 2017-01-27 22:54:43
Connection ID: CLI
Src IP: 172.16.4.193
Dst IP: 194.87.234.129
Src Port: 49202
Dst Port: 80
OS Fingerprint: 172.16.4.193:49202 - Windows XP/2000 (RFC1323+, w+, tstamp-) [GENERIC]
OS Fingerprint: Signature: [8192:128:1:52:M1460,N,W8,N,N,S:.:Windows:?]
OS Fingerprint: -> 194.87.234.129:80 (distance 0, link: ethernet/modem)
SRC: GET /?ct=Vivaldi&biw=Vivaldi.95ec76.406i7c5k7&oq=h8fItKeRVawGyjRaFcw1nyYdeAwgQ8_qtiEKBzBKfgZ6D-hyMZAh1z6LRVvQ42w&tuif=2320&q=wH7QMvXcJwDNFY
bGMvrER6NbNknQA0KPxpH2_drZdZqxKGni2Ob5UUSk6FqCEh3&yus=Vivaldi.114tq57.406t1v7x8&br_fl=4180
SRC: ACCEPT: text/html, application/xhtml+xml, */*
SRC: REFERER: http://www.homeimprovement.com/remodeling-your-kitchen-cabinets.html
SRC: ACCEPT-LANGUAGE: en-US
SRC: USER-AGENT: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
SRC: ACCEPT-ENCODING: gzip, deflate
SRC: HOST: tyu.benme.com
SRC: CONNECTION: Keep-Alive
DST: 200 OK
DST: SERVER: nginx/1.6.2
DST: DATE: Fri, 27 Jan 2017 22:54:38 GMT
DST: CONTENT-TYPE: text/html;charset=UTF-8
DST: CONTENT-LENGTH: 1842
DST: CONNECTION: keep-alive
DST: VARY: Accept-Encoding
DST: CONTENT-ENCODING: gzip
DST: <!DOCTYPE html>\x0a<html lang="en">\x0a<head>\x0a <title></title>\x0a <meta charset="UTF-8">\x0a <meta http-equiv="X-UA-Compatible" content="IE=EDGE">\x0a <
meta name="apple-mobile-web-app-capable" content="yes">\x0a <meta name="apple-mobile-web-app-status-bar-style" content="black">\x0a <meta name="viewport" content
```

Figure 7: Results in the CapME! window

Source: Compiled by the authors based on application CapME!

We examined the first block of blue text, and we came to the following conclusions:

- The figure shows that this is a request to the target web server.
- It was observed that there are two URLs in this block.
- The first is tagged with SRC: REFERER.
- This indicates the web page that was first accessed by the source computer. However, the server redirected the browser's HTTP GET request to SRC:HOST, which indicated that something in the HTML had sent the source to this site, which indicated that this was most likely a drive-by attack.
- The user intends to connect to www.homeimprovement.com
- The browser refers the user to ty.benme.com URL
  - According to the referred URL, we checked what kind of content is requested by the source host from tybenme.com? We examined the DST server block of the transcript too.
  - The content is in gzip format, which could easily be a malware executable that, once downloaded, runs some malicious code on the host. Because of the compression of the file, its contents are obscured, so it is not easy to determine what is in the file.

CapME! allows checking in the HTTP dashboard section which web pages were visited during the period we are analysing. The HTTP Sites section of the dashboard provides the necessary information. In our case, the following website data were identified:

- www.bing.com
- p27dokhpz2n7nvgr.1jw2lx.top
- homeimprovement.com
- tyu.benme.com
- www.google-analytics.com
- api.blockcipher.com
- spotsbill.com
- fpdownload2.macromedia.com
- retrotip.visionurbana.com.ve

Some of the above websites were already known from earlier activities, not all of which were involved in the exploitation activities. Each URL was searched for on the Internet, and the URLs were enclosed in quotes in the search. In none of these cases were we directly linked to the website. The following conclusions were drawn from the investigations:

- These sites are likely part of the exploit campaign:
  - p27dokhpz2n7nvgr.1jw2lx.top
  - homeimprovement.com
  - tyu.benme.com
  - spotsbill.com
  - retrotip.visionurbana.com.ve
- The HTTP – MIME Types are in the Tag Cloud:
  - image/jpeg
  - text/plain
  - text/html
  - image/gif
  - image/png
  - application/javascript
  - application/x-shockwave-flash
  - text/json

4. Investigate the Exploit with Sguil

Sguil is a network security analyst tool. Sguil ensures access to real-time events, data and raw packets. In addition, Sguil helps with the Network Security Monitoring and event-driven analysis processes and procedures. The program is written in tcl/tk by Robert “Bamm” Visscher. Tcl is the Tool Command Language, an interpreted programming language. It was designed for fast software development.

Tk is a graphical part that draws the data on an analyst's screen. Sguil applies the following tools:

- Snort: this provides alert data, which is why it is very popular in the incident management field. The second copy of Snort collects full content data.
- The keepstats option of the Snort stream4 preprocessor: using this, Sguil receives TCP-based session data.
- Tcpflow: this regenerates the full content trace files to display the application data.
- P0f: it profiles traffic to identify operating system fingerprints.
- MySQL stores alert and packet data collected from Snort.[10]

Sguil can also help to analyse IDS alerts and gather additional information about the sequence of events related to the attack. Among other things, Sguil can help to identify the time when the event occurred, which in our case was:

- According to Sguil, the timestamps for the first and last of the alerts that occurred within about a second of each other are 22:54:42 to 22:55:28. The entire exploit occurred in less than a minute.

Other options include checking the field information in the packet header and the IDS signature rules associated with the alert to determine which malware caused the alert, which can be very helpful in future recovery processes. In our case, the following result was obtained:

- According to the IDS signature rule, Malware_family PseudoDarkLeech malware family triggered this alert.

During the rest of the analysis, event messages were checked for each alert ID associated with the attack, which returned the following results:

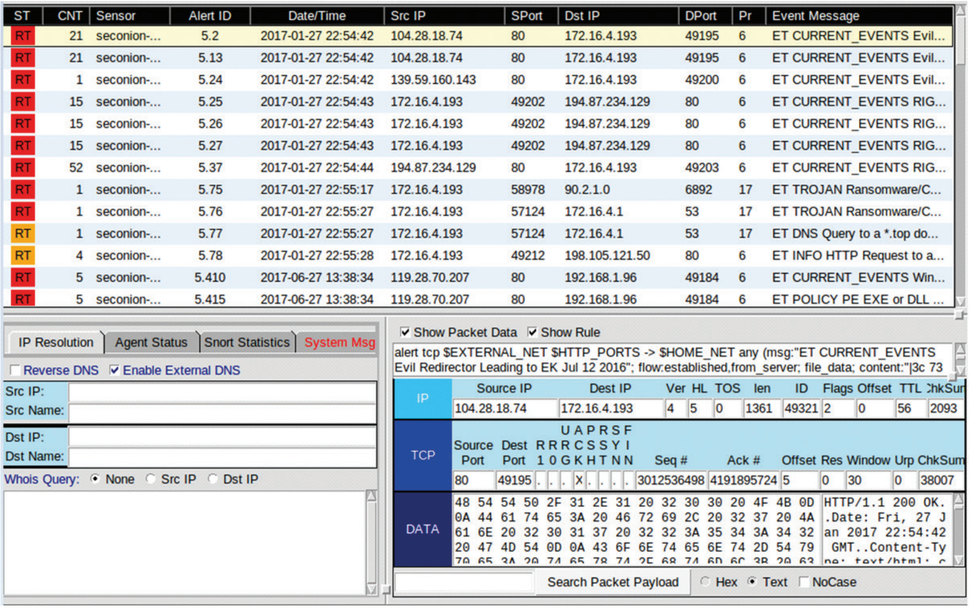
- According to the Event Messages in Sguil, the RIG EK Exploit exploit kit is involved in this attack.

Beyond labelling the attack as trojan activity, other information is provided regarding the type and name of the malware:

- ransomware
- Cerber

Based on the alerts so far, it appears that the basic vector of the attack was a visit to a malicious website.

[10] Bejtlich 2010



ST	CNT	Sensor	Alert ID	Date/Time	Src IP	SPort	Dst IP	DPort	Pr	Event Message
RT	21	seconion-...	5.2	2017-01-27 22:54:42	104.28.18.74	80	172.16.4.193	49195	6	ET CURRENT_EVENTS Evil...
RT	21	seconion-...	5.13	2017-01-27 22:54:42	104.28.18.74	80	172.16.4.193	49195	6	ET CURRENT_EVENTS Evil...
RT	1	seconion-...	5.24	2017-01-27 22:54:42	139.59.160.143	80	172.16.4.193	49200	6	ET CURRENT_EVENTS Evil...
RT	15	seconion-...	5.25	2017-01-27 22:54:43	172.16.4.193	49202	194.87.234.129	80	6	ET CURRENT_EVENTS RIG...
RT	15	seconion-...	5.26	2017-01-27 22:54:43	172.16.4.193	49202	194.87.234.129	80	6	ET CURRENT_EVENTS RIG...
RT	15	seconion-...	5.27	2017-01-27 22:54:43	172.16.4.193	49202	194.87.234.129	80	6	ET CURRENT_EVENTS RIG...
RT	52	seconion-...	5.37	2017-01-27 22:54:44	194.87.234.129	80	172.16.4.193	49203	6	ET CURRENT_EVENTS RIG...
RT	1	seconion-...	5.75	2017-01-27 22:55:17	172.16.4.193	58978	90.2.1.0	6892	17	ET TROJAN Ransomware/C...
RT	1	seconion-...	5.76	2017-01-27 22:55:27	172.16.4.193	57124	172.16.4.1	53	17	ET TROJAN Ransomware/C...
RT	1	seconion-...	5.77	2017-01-27 22:55:27	172.16.4.193	57124	172.16.4.1	53	17	ET DNS Query to a *.top do...
RT	4	seconion-...	5.78	2017-01-27 22:55:28	172.16.4.193	49212	198.105.121.50	80	6	ET INFO HTTP Request to a...
RT	5	seconion-...	5.410	2017-06-27 13:38:34	119.28.70.207	80	192.168.1.96	49184	6	ET CURRENT_EVENTS Win...
RT	5	seconion-...	5.415	2017-06-27 13:38:34	119.28.70.207	80	192.168.1.96	49184	6	ET POLICY PE EXE or DLL ...

☐ Reverse DNS ☑ Enable External DNS

Src IP:
Src Name:
Dst IP:
Dst Name:

Whois Query: ● None ○ Src IP ○ Dst IP

☑ Show Packet Data ☑ Show Rule

```
alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any (msg:"ET CURRENT_EVENTS
Evil Redirector Leading to EK Jul 12 2016"; flow:established,from_server; file_data; content:"|3c 73
```

IP	Source IP	Dest IP	Ver	HL	TOS	len	ID	Flags	Offset	TTL	ChkSum
	104.28.18.74	172.16.4.193	4	5	0	1361	49321	2	0	56	2093

TCP	Source Port	Dest Port	R 1	R 0	U R G	A C K	P S H	R S T	S Y N	F I N	Seq #	Ack #	Offset	Res	Window	Urp	ChkSum
	80	49195	.	.	.	X	.	.	.	.	3012536498	4191895724	5	0	30	0	38007

DATA

```
48 54 54 50 2F 31 2E 31 20 32 30 30 20 4F 4B 0D   HTTP/1.1 200 OK.
0A 44 61 74 65 3A 20 46 72 69 2C 20 32 37 20 4A   .Date: Fri, 27 J
61 6E 20 32 30 31 37 20 32 32 3A 35 34 3A 34 32   an 2017 22:54:42
20 47 4D 54 0D 0A 43 6F 6E 74 65 6E 74 2D 54 79    GMT..Content-Ty
70 65 3A 20 74 65 78 74 2F 68 74 6D 6C 3B 20 63   pe: text/html; c
```

○ Hex ● Text ☐ NoCase

Figure 8: Listed IDS alerts is Sguil

Source: Compiled by the authors based on application Sguil.

4.1. Transcripts of events

After selecting the ID number of the alert shown in the picture above, it is possible to retrieve the Transcript, which can provide us with additional useful information. Examples include the sending and host websites, the browser or search engine used. The transcript may also show specific information such as the type of request, the file's name, format, or website address. For example, during our investigation, we were able to identify the following information from one of our alerts:

- HTTP/1.1 GET request kind of request was involved
- dle_js.js is files requested
- the referer website was www.homeimprovement.com/remodeling-your-kitchen-cabinets.html and the host website was retrotip.visionbura.com.ve
- the content encoded by gzip

By examining a recent alert, we could identify much more detailed data, making it easier to identify indicators of compromise (IoC), analyse individual incidents, and even help improve the recovery process. These results were:

- 3 requests and 3 responses were involved in this alert
- GET /?ct=Vivaldi&biw=Vivaldi.95ec was the first request
- www.homeimprovement.com/remodeling-your-kitchen-cabinets.html was the referrer
- tyu.benme.com was the host server request to

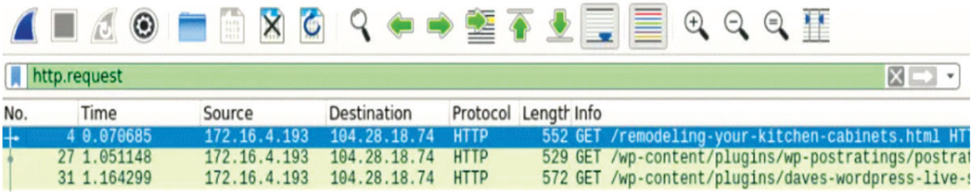
- POST /?oq=CEh3h8…. Vivaldi was the second request
- tyu.benme.com was the host server request to
- the response was encoded by gzip
- GET /?biw=SeaMonkey.105…. was the third request
- http://tyu.benme.com/?biw…was the referrer
- application/x-shockwave-flash was the Content-Type of the third response
- CWS was the first three characters of the data in the response; the data starts after the last DST: entry; CWS is a file signature; file captions help identify the type of file representing different data types
- swf file was downloaded, Adobe Flashuses this type of file

5. Use Wireshark to investigate an attack

In the following analysis, Wireshark was used to examine the details of the attack. Wireshark is a network protocol analysis program that can capture packets from a network connection. In networking, a packet is a small segment of a larger message. Data transmitted over computer networks such as the Internet is divided into packets. The receiving computer or device then reassembles these packets. Wireshark is one of the most popular sniffer software that does three things:

- capture packets: listens to network connections in real-time and then collects the entire traffic stream
- filtering: this software has a slicing and dicing function for the captured information; if someone needs only specific information, it is available with filtering
- visualisation: it is possible to inspect and measure network packets; visualisation capability is available for complete conversations on the network[11]

In Sguil, we pivoted to select Wireshark from the menu for the chosen alarm ID. The pcap for the alert was opened in Wireshark. By default, Wireshark uses relative time per packet, which is not useful enough to isolate the exact time an event occurred. To make this more detailed, it is possible to select a time display format based on seconds, which makes it much easier to identify the exact time of the event. Several additional filtering options are available in Wireshark; in our analysis, we used the http.request display filter to filter only web requests, illustrated in Figure 9.


http.request

```
No.     Time        Source         Destination    Protocol Lengtt Info
      4 0.070685    172.16.4.193   104.28.18.74   HTTP        552 GET /remodeling-your-kitchen-cabinets.html HT
     27 1.051148    172.16.4.193   104.28.18.74   HTTP        529 GET /wp-content/plugins/wp-postratings/postrat
     31 1.164299    172.16.4.193   104.28.18.74   HTTP        572 GET /wp-content/plugins/daves-wordpress-live-
```

Figure 9: http.request requests in Wireshark

Source: Compiled by the authors based on application Wireshark.

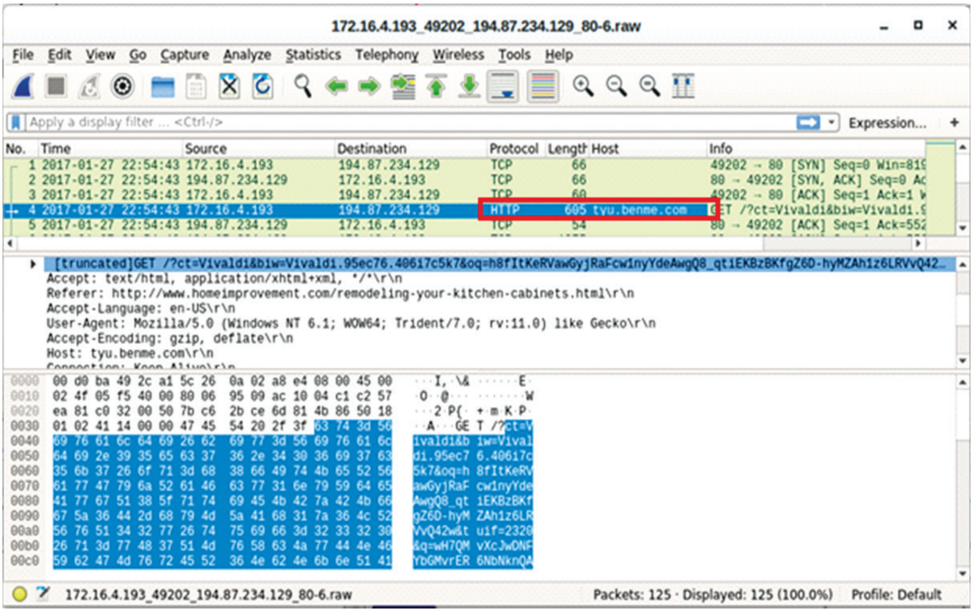
[11] CompTIA 2020

We have selected the first package. In the packet details area, we expanded the Hypertext Transfer Protocol application layer data, which was used to determine to which website the web page of the search engine redirected the user. This is also an important piece of information for further analysis.

5.1. View HTTP objects

As shown above, several HTTP objects were involved in the attack, so their analysis is of paramount importance. To do this, it is possible to extract and save the remodeling. html page in Wireshark, which will provide us with a copy of the page we originally wanted to access. Afterward, in Sguil, we can check, among other things, which file the http request was for and which web page it pointed to. In the case we examined, the http request was for a JavaScript file named dle_js.js, and the host server was retrotip.visionurbana.com.ve.

In the application, the display filter function allows displaying information showing which specific requests are associated with the alert. For example, if we assign the Host column to the information displayed, the application will show us which page the infected website redirects us to. In our case, this is illustrated in Figure 10 where we can see that the Host address is tyu.benme.com. Such information can be the basis for further analysis and can therefore be of significant value.



```
No. Time                Source          Destination     Protocol Length Host           Info
  1 2017-01-27 22:54:43 172.16.4.193    194.87.234.129  TCP      66                    49202 → 80 [SYN] Seq=0 Win=819
  2 2017-01-27 22:54:43 194.87.234.129  172.16.4.193    TCP      66                    80 → 49202 [SYN, ACK] Seq=0 Ac
  3 2017-01-27 22:54:43 172.16.4.193    194.87.234.129  TCP      60                    49202 → 80 [ACK] Seq=1 Ack=1 W
  4 2017-01-27 22:54:43 172.16.4.193    194.87.234.129  HTTP     605    tyu.benme.com  GET /?ct=Vivaldi&biw=Vivaldi.9
  5 2017-01-27 22:54:43 194.87.234.129  172.16.4.193    TCP      54                    80 → 49202 [ACK] Seq=1 Ack=552
```

```
  [truncated]GET /?ct=Vivaldi&biw=Vivaldi.95ec76.40617c5k7&oq=h8fItKeRVawGyjRaFcw1nyYdeAwgQ8_qtiEKBzBKfgZ6D-hyMZAh1z6LRVvQ42…
  Accept: text/html, application/xhtml+xml, */*\r\n
  Referer: http://www.homeimprovement.com/remodeling-your-kitchen-cabinets.html\r\n
  Accept-Language: en-US\r\n
  User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko\r\n
  Accept-Encoding: gzip, deflate\r\n
  Host: tyu.benme.com\r\n
```

```
0000  00 d0 ba 49 2c a1 5c 26  0a 02 a8 e4 08 00 45 00   ···I,·\&  ······E·
0010  02 4f 05 f5 40 00 80 06  95 09 ac 10 04 c1 c2 57   ·O··@···  ·······W
0020  ea 81 c0 32 00 50 7b c6  2b ce 6d 81 4b 86 50 18   ···2·P{·  +·m·K·P·
0030  01 02 41 14 00 00 47 45  54 20 2f 3f 63 74 3d 56   ··A···GE  T /?ct=V
0040  69 76 61 6c 64 69 26 62  69 77 3d 56 69 76 61 6c   ivaldi&b  iw=Vival
0050  64 69 2e 39 35 65 63 37  36 2e 34 30 36 69 37 63   di.95ec7  6.406i7c
0060  35 6b 37 26 6f 71 3d 68  38 66 49 74 4b 65 52 56   5k7&oq=h  8fItKeRV
0070  61 77 47 79 6a 52 61 46  63 77 31 6e 79 59 64 65   awGyjRaF  cw1nyYde
0080  41 77 67 51 38 5f 71 74  69 45 4b 42 7a 42 4b 66   AwgQ8_qt  iEKBzBKf
0090  67 5a 36 44 2d 68 79 4d  5a 41 68 31 7a 36 4c 52   gZ6D-hyM  ZAh1z6LR
00a0  56 76 51 34 32 77 26 74  75 69 66 3d 32 33 32 30   VvQ42w&t  uif=2320
00b0  26 71 3d 77 48 37 51 4d  76 58 63 4a 77 44 4e 46   &q=wH7QM  vXcJwDNF
00c0  59 62 47 4d 76 72 45 52  36 4e 62 4e 6b 6e 51 41   YbGMvrER  6NbNknQA
```

172.16.4.193_49202_194.87.234.129_80-6.raw
Packets: 125 · Displayed: 125 (100.0%)
Profile: Default

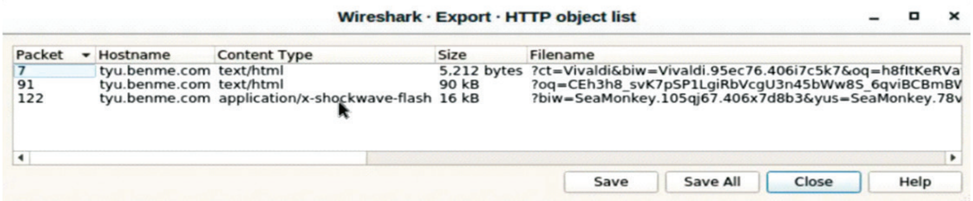
Figure 10: Redirected target address

Source: Compiled by the authors based on application Sguil.

6. Create a hash for an exported malware file

The investigation identified which website the user was trying to access, but the page redirected the user to another website, resulting in files being downloaded to the user's computer from a potentially malicious website. To determine whether a malicious file has been downloaded, it is possible to analyse the hash of the file. For this purpose, we primarily need the downloaded file and can analyse it using, for example, the VirusTotal website.

To generate the hash, it is needed first to export the files from the HTTP object list in Wireshark. In the case of our test, this was two text/html files and an application/x-shockwave-flash file, shown in Figure 11.



Wireshark · Export · HTTP object list

Packet	Hostname	Content Type	Size	Filename
7	tyu.benme.com	text/html	5,212 bytes	?ct=Vivaldi&biw=Vivaldi.95ec76.406i7c5k7&oq=h8fItKeRVa
91	tyu.benme.com	text/html	90 kB	?oq=CEh3h8_svK7pSP1LgiRbVcgU3n45bWw8S_6qviBCBmBV
122	tyu.benme.com	application/x-shockwave-flash	16 kB	?biw=SeaMonkey.105qj67.406x7d8b3&yus=SeaMonkey.78v

Figure 11: HTTP object list in Wireshark

Source: Compiled by the authors based on application Wireshark.

From the available files, a hash can be generated, which can then be entered into VirusTotal to see if this hash is present in its database for any previously discovered malicious code. VirusTotal is a website developed by Hispasec Sistemas in June 2004. VirusTotal makes fusions of many scan engines and antivirus software. The goal is to examine viruses missed on the host-based antivirus software. Files can be uploaded to the website or sent via email. In addition, there is suspect URL scan capability and VirusTotal dataset search. VirusTotal scans items with more than 70 antivirus and URL/domain blocklist services to extract signs from the examined content.

VirusTotal has got several file submission mechanisms and techniques. The web interface has the principal scanning priority among the publicly available submission methods. Uploaded files can be scripted in any programming language using the HTTP-based public API. These files are shared with the examining part and the sender as well. If somebody uploads a suspicious file or URL, that sender raises global IT security because VirusTotal is using it to develop their database with these kinds of data. Additional functions make the VirusTotal up to date, sharing based on database and community. For example, it allows users to comment on files and URLs and share their experiences with each other.

Some scanners identify everyday items as malicious. In these cases, the VirusTotal can separate and recognise the differences between malicious content and false positives. VirusTotal for dynamic analysis of malware uses the Cuckoo sandbox.

VirusTotal can provide the detection label information. Like the URL scanners cases, which separate malware sites, phishing sites, suspicious sites, etc.

The application shows us much information that can be useful for further analysis, such as MD5, SHA-1 and SHA-256 hash values, time of the first occurrence, time of last confirmation and analysis, file type and the names of infected files detected so far. For the analysed application/x-shockwave-flash file, the following results were obtained:

- MD5: f858070326067ba282d2a63969868e5a
- SHA-1: 97a8033303692f9b7618056e49a24470525f7290
- SHA-256: b3669ec83fb4bba5257da8c68b32dc15d1a08e9e8c22c7483698f29de2839b5f
- File type: Flash
- File size: 15.88 KB (16261 bytes)
- First Seen in The Wild: 2017-01-27 22:39:08 UTC
- First Submission: 2017-01-27 22:41:40 UTC
- Last Submission: 2022-07-28 02:22:14 UTC
- Last Analysis 2022-06-22 10:43:06 UTC
- At the time of analysis, 22 different names for the malicious code were found by the application, and 30 out of 56 antivirus programs had rules to identify this hash as a malicious file.

Essentially, this provides us with all the information we need to prevent this type of malicious code from entering our systems in the future.

7. Examine exploit artefacts

Further examination of the HTML code extracted from Wireshark provides a wealth of additional useful information. For example, it is possible to determine which scripts in the header may contain malicious activities. In the header of the code we examined, there was a script section that loads the JavaScript file dle_js.js. In addition, the iFrame that loads the content from tyu.benme.com is defined in the HTML body. The script was the following:
<script type="text/javascript" src="//retrotip.visionurbana.com.ve/engine/classes/js/dle_js.js"*></script>*

Inspection of the dle_js.js file in a text editor of choice showed that Javascript document.write() writes the content to the web page, creating an iFrame that redirects to the tyu.benme.com URL. The iFrame was the following:
<iframe src="https://tyu.benme.com/?q=zn_QMvXcJwDQDofGMvrESLtEMUbQA0KK2OH_76iyEoH9JHT1vrTUSkrttgWC&biw=Amaya.81lp85.406f4y5l9&oq=elTX_fUlL7ABPAuy2EyALQZnlY0IU1IQ8fj630PWwUWZ0pDRqx29UToBvdeW&yus=Amaya.110oz60.406a7e5q8&br_fl=4109&tuif=5364&ct=Amaya" *width=290 height=257 ></ifr' +'ame>*

The exploit kit we found is also an automated software to exploit known vulnerabilities in systems or programs. Attackers use them while victims are surfing on the web. Meanwhile, in web surfing, the main purpose is for the victim to download and execute some variable of malware. Therefore, it is very difficult to determine the real fact of the attack because the exploit kits work in the background. Of course, avoiding unknown links and maintaining the software up to date can ensure a higher level of protection against an attack situation.[12]

An exploit kit is a package to hand over malware. If a victim's computer has a determined number of vulnerabilities, in that case, if the user reaches compromised URLs, it can happen when an exploit is delivered. For this purpose, the malware will be executed on that computer. A software vulnerability is a flaw in software that allows an attacker to take control of the system. These flaws can result from how the software is designed or coding errors. The attacker first determines whether a software vulnerability exists by scanning the system. From the scan, the attacker can find out what types of software are present on the system, whether they are up-to-date, and whether any software packages are vulnerable. If the attacker can find this out, he will have a better idea of the types of attacks he can launch against the system. A successful attack would result in the attacker being able to execute malicious commands on the target system.[13]

Known vulnerabilities are named in a reference list of Common Vulnerabilities and Exposures (CVE). The Common Vulnerabilities and Exposures (CVE) list is a dictionary that has created a common, standardised naming convention for system, network and software vulnerabilities to enable organisations to share information about new risks and establish baselines for assessing the effectiveness of cybersecurity tools and services. The CVE aims to facilitate sharing information on known vulnerabilities so that cybersecurity strategies can be updated to consider the latest security flaws and security issues. Common targets for exploits are popular software with many known vulnerabilities, such as Adobe Flash, Oracle Java and Internet Explorer.[14]

The first step is to make contact with the victim. For example, attackers often use spammed email, and social engineering lures to get individuals to click on a link to an exploit kit server. Another example is when the victim visits a compromised website and clicks on malicious advertising.

The remaining victims are redirected to an alternative landing page which is no longer the real URL. Code embedded into this landing page then proceeds to determine if the victim's device has any vulnerable browser-based applications that correspond to the exploits in the kit. If there are no vulnerabilities, the attack will be stopped. The website will send traffic to the exploit if the vulnerability is detected.[15]

The victims are subsequently redirected to the landing page of the exploit kit. It defines which vulnerabilities will be exploited during the attack. The mode of the exploit is carried out and is determined by the software. If web browsers are the target, the exploit will take the form of code embedded within the web page. The exploits are the first thing supplied to the victim's browser. These exploits will make use of previously known flaws.

[12] Qin et al. 2016

[13] JFrog 2021

[14] Horváth 2020

[15] Tutorialspoint 2022

Malware is executed on the victim's computer after successful exploitation. To the extent of the effect of the malware, there are many different scenarios. Exploit kits can be used to spread several types of malwares. In our case, we reached the homeimprovement.com URL, which had a malicious iFrame that redirected us to a malicious site. This site (tyu.benme.com) executed the malicious JavaScript ad and delivered the malicious Adobe Flash file.

8. Summary

Exploit Kits are digital weapons that are often used by cybercriminals. For example, EK automatically infected malware on the victim's computer without knowing facts by exploiting vulnerabilities.

In the malware exploitation scenario, the victim searched for a page on a search engine site and reached the URL of an infected website, which was identified as a compromised website according to the analysis of the detailed results. The exploit backend contacted the webserver to execute malicious JavaScript code. The EK backend then communicated with the web server and delivered the malicious JavaScript URL to the victim. As a result, ransomware malware was sent in a way the victim did not detect.

The data-driven analysis of the malware exploit contains several similar or identical process steps that could be used or researched in the event of a malware attack against a military network. The stages can provide useful data analysis techniques and methods that can help understand the behaviour of the malicious malware. In particular, malicious embedded URL encoders. It can also contribute to increasing the level of cybersecurity of the government or even military and increasing information and cyber awareness of government and defence sector personnel using external links from the Internet.

References

Azarmi, Bahaaldine (2017): *Learning Kibana 5.0.* Birmingham: Packt Publishing.

Bejtlich, Richard (2010): *The Tao of Network Security Monitoring. Beyond Intrusion Detection.* Boston: Addison-Wesley Professional.

CompTIA: What Is Wireshark and How Is It Used? *CompTIA,* 10 November 2020. Online: www.comptia.org/content/articles/what-is-wireshark-and-how-to-use-it

Fang, Yufei – Shan, Zhiguang – Wang, Wei (2021): Modeling and Key Technologies of a Data-Driven Smart City System. *IEEE Access,* 9, 91244–91258. Online: https://doi.org/10.1109/ACCESS.2021.3091716

GoLinuxCloud: *ELK Stack: Configure elasticsearch cluster setup CentOS/RHEL 7/8.* 2020. Online: www.golinuxcloud.com/setup-configure-elasticsearch-cluster-7-linux/

Horváth, Ingrid: Understanding Common Vulnerabilities and Exposures. *Invensis,* 17 September 2020. Online: www.invensislearning.com/blog/understanding-common-vulnerabilities-and-exposures/

JFrog: What is a Software Vulnerability? *JFrog,* 22 August 2021. Online: https://jfrog.com/knowledge-base/software-vulnerability/

Jia, Kunqi – Wang, Zhihua – Fan, Shuai – Xiao, Jucheng – He, Guangyu (2018): Data-Driven Architecture Design and Application of Power Grid Cyber Physical System. *Power System Technology,* 42(10), 3116–3127. Online: https://doi.org/10.13335/j.1000-3673.pst.2018.0876

O'Driscoll, Aimee: What is an exploit kit (with examples) and how do cybercriminals use them? *Comparitech,* 07 May 2019. Online: www.comparitech.com/blog/information-security/exploit-kits/

Qin, Feng – Liu, Dongxia – Sun, Bingda – Ruan, Liu – Ma, Zhanhong – Wang, Haiguang (2016): Identification of Alfalfa Leaf Diseases Using Image Recognition Technology. *Public Library of Science,* 11(12), 1–7. Online: https://doi.org/10.1371/journal.pone.0168274 ; DOI: https://doi.org/10.1371/journal.pone.0168274

Tutorialspoint: What is an Exploit Kit? (Stages, Process, How to Stay Safe). *Tutorialspoint,* 14 June 2022. Online: www.tutorialspoint.com/what-is-an-exploit-kit-stages-process-how-to-stay-safe

Wang, Ying – Li, Peilong – Jiao, Lei – Su, Zhou – Cheng, Nan – Shen, Xuemin Sh. – Zhang, Ping (2017): A Data-Driven Architecture for Personalized QoE Management in 5G Wireless Networks. *IEEE Wireless Communications,* 24(1), 102–110. Online: https://doi.org/10.1109/MWC.2016.1500184WC

Wang, Zhihua – Xiao, Jucheng – Jia, Kunqi – Gao, Feng – Tang, Yuanhe – He, Guangyu (2018): A Data-Driven Architecture Design of Stream Computing for the Dispatch and Control System of the Power Grid. 2nd *IEEE Conference on Energy Internet and Energy System Integration (EI2),* 1–6. Online: https://doi.org/10.1109/EI2.2018.8582404